1 Matrices and rank (our tools today)

Look at these two matrices:

\[
\begin{pmatrix}
0 & 1 & 0 \\
0 & 0 & 0 \\
1 & 1 & 0
\end{pmatrix}
\quad \begin{pmatrix}
0 & 1 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0
\end{pmatrix}
\]

The determinant gives us the same answer (both nonsingular) but the one on the left “looks reacher”.

The \textbf{rank} of a matrix is the maximum number of rows (or columns) that are linearly independent.

Keep in mind the following:

- It is not obvious that the number of independent rows equals the number of independent columns (especially for non-square matrices);
- The rank depends on the underlying field, so we should write $\text{rank}_F(A)$ when the matrix $A \in F^{n \times m}$. For instance this matrix
can be singular over $\mathbb{F}_2$, and nonsingular over $\mathbb{Q}$.

In the sequel we simply write $\text{rank}(A)$ when the result does not depend on a particular field, or when the field has being specified at the beginning.

\[
\text{rank}(A + B) \leq \text{rank}(A) + \text{rank}(B) \quad (1)
\]

Today we solve a problem using this rank inequality (Section 2) and show these properties of the rank (Section A).

## 2 Jigsaw puzzle with graphs

You have to cover all edges of the complete graph on $n$ nodes using as few complete bipartite graphs as possible, and every edge must be covered exactly once.

What is the minimum number $m = m(n)$ of bipartite graphs that we must use?

**Exercise 1.** Show that $m(n) \leq n - 1$.

**Exercise 2.** Suppose we allow edges to be covered more than once. Prove that $\lceil \log_2 n \rceil$ bipartite graphs suffice, in this case.

### 2.1 First lower bound

For this problem it is a good idea to work over the field $\mathbb{F} = \mathbb{Q}$.

*Maybe you can check what happens to the proof when $\mathbb{F} = \mathbb{F}_2$.\)
Consider the adjacency matrix of a bipartite graph, say $B_i$. The adjacency matrix of the complete graph is $K$. The rank is the key:

$$\text{rank}(B_i) = 2 \quad \text{and} \quad \text{rank}(K) = n$$ \hfill (2)

**Exercise 3.** Prove these two identities.

Since every edge is covered exactly once

$$K = B_1 + B_2 + \cdots + B_m$$

and (1) implies

$$n = \text{rank}(K) \leq \text{rank}(B_1) + \cdots + \text{rank}(B_m) = 2m$$

which implies our first lower bound:

You need $m(n) \geq n/2$ bipartite graphs.

### 2.2 Improved lower bound

We want to replace the matrices $B_i$’s with even “simpler” matrices that have rank 1. We **orient** the edges of each bipartite graph from one side to the other (no matter which one):

bipartite complete graph $X_k \cup Y_k \implies$ matrix $A_k$

and $A_k$ has 1 in column $i$ and row $j$ iff $i \in X_i$ and $j \in Y_i$ (and 0 otherwise).

This picture shows how a matrix $B_k$ changes into a matrix $A_k$ (dark area represent 1’s):

---

**Exercise 4.** Prove that $\text{rank}(A_k) = 1$.

**Exercise 5** (wrong proof & and wrong bound). We have that

$$K = A_1 + \cdots + A_m$$

which implies

$$n = \text{rank}(K) = \text{rank}(A_1 + \cdots + A_m) \leq \text{rank}(A_1) + \cdots + \text{rank}(A_m) = m$$

Find the mistake in this proof, and explain why this bound cannot be true.
Correct proof. Consider the matrix
\[ S \triangleq A_1 + \cdots + A_m \]
We show that
\[ \text{rank}(S) \geq n - 1 \]
which then implies a tight bound \( m(n) = n - 1 \).

Every edge of the complete graph is covered in “exactly one direction” by our matrices \( A_k \). That is, for every \((i, j)\), either \( S_{ij} = 1 \) or \( S_{ji} = 1 \). Translated in the matrix language
\[ S + S^T = K = J - I \quad (3) \]
with \( J \) being the matrix with all 1’s and \( I \) the identity matrix (both \( n \times n \)).

First trick: \( Sx = 0 \implies x^T(S + S^T)x = 0 \)

Here is the proof:
\[ x^T(S + S^T)x = x^T(S^T x + Sx) = x^T(S^T x + 0) = x^T S^T x = 0^T x = 0. \]

By contradiction suppose \( \text{rank}(S) \leq n - 2 \).

Second trick: \( \text{rank}(S) \leq n - 2 \implies Sx = 0 \) with \( \sum_i x_i = 0 \) for some \( x \neq 0 \)

To prove this recall that (for any \( m \times n \) matrix \( A \))
\[ \text{rank}(A) \leq n - 1 \implies Ax = 0 \text{ for some } x \neq 0 \]
If we add to \( S \) one row with all 1’s, the resulting matrix \( A \) has \( \text{rank}(A) \leq \text{rank}(S) + 1 \leq n - 1 \). Therefore \( Ax = 0 \) for some \( x \neq 0 \). The added row \( 1 \triangleq (1, \ldots, 1) \) expresses the condition \( \sum_i x_i = 0 \).

Put the things together

Since \( \sum x_i = 0 \) we have \( Jx = 0 \) and thus (3) gives us
\[ 0 = x^T(S + S^T)x = x^T K x = x^T(J - I)x = x^T(-x) = -||x||^2 \neq 0 \]
This contradiction proves \( \text{rank}(S) \geq n - 1 \). (Note: the last inequality, \(-||x||^2 \neq 0\), holds because we work with the field \( \mathbb{F} = \mathbb{Q} \).)
2.3 The origin of this problem (not in this lecture)

Routing over an hypercube is very simple: look at the labels and move to the neighbor that reduces your distance to the destination. For instance, to go from 001 to 111 we follow the path

$$001 \rightarrow 101 \rightarrow 111$$

and the path is the shortest path in the graph.

I would like to put \( \{0, 1\} \)-address (a binary string) on the nodes of any graph and do routing in the same way. For the triangle, I’m in trouble:

No matter what I write on the top node, two edges will be at distance 2 and routing will fail (try with “??? = 010”). The solution is to introduce a jolly-joker “∗”, meaning that two coordinates are different only if one is 0 and the other one is 1:

If you label the edges by the coordinate in which the addresses of the endpoints are different, you should see the connection between this problem and our “jigsaw puzzle with graph” (see the picture at page 1).

3 Subspace, Basis, and Dimension

In the sequel \( V \) is a linear space of finite dimension, that is, there exist a finite subset of elements \( v_1, \ldots, v_N \in V \) such that \( V = \text{span}(v_1, \ldots, v_N) \).

A subspace of \( V \) is a subset \( U \subseteq V \) such that the addition and scalar product of \( V \) make \( U \) into a linear space.
A basis for $U$ is a subset $u_1, \ldots, u_k \in V$ such that

$$U = \text{span}(u_1, \ldots, u_k)$$

$u_1, \ldots, u_k$ are linearly independent

In this case we say that $U$ has dimension $k$,

$$\text{dim}(U) = k$$

**Remark 1.** Note that any two basis for $U$ must have the same cardinality. If $u_1, \ldots, u_k$ is a basis for $U$ then every $u_i$ must be in $U$. *(Exercise!)*

### 3.1 Orthogonality

Consider the linear space of all vectors of length $n$ over some field $F$

$$V = F^n$$

and the standard **inner product** between two such vectors

$$u \cdot v \triangleq u_1v_1 + \cdots + u_nv_n$$

(multiplications and additions over $F$.)

We say that $u$ and $v$ are **orthogonal** if their inner product is 0:

$$u \perp v \iff u \cdot v = 0$$

The **orthogonal complement** of a subset $U \subseteq V$ is

$$U^\perp \triangleq \{v \in V | \text{ for all } u \in U, \ v \perp u\}$$

**Exercise 6.** Show that $U^\perp$ is always a subspace of $V$.

---

**Rank Theorem**

**Theorem 1.** For every subspace $U$ of $V$

$$\text{dim}(U) + \text{dim}(U^\perp) = n \quad \text{where } n = \text{dim}(V)$$

We do not prove this theorem in this lecture.
3.1.1 Application to eventown

Recall from lecture 1:

**Eventown Problem:** All clubs must have **even** cardinality, their pairwise intersection must be even as well, and no two clubs can have the same members: we want distinct subsets $S_1, \ldots, S_m \subseteq [n]$ such that

\begin{align}
|S_i| & \text{ is even for all } i \\
|S_i \cap S_j| & \text{ is even for all } i \neq j
\end{align}

What is the maximum number $m = m(n)$ of such subsets?

We have seen in the first lecture a simple construction of $2^{\lfloor n/2 \rfloor}$ such subsets, that is $m(n) \geq 2^{\lfloor n/2 \rfloor}$. We now prove that this is the best possible

**Theorem 2.** The maximum number of clubs in Eventown is $m(n) = 2^{\lfloor n/2 \rfloor}$.

**Proof.** Consider the incidence vectors $v_1, \ldots, v_m$ corresponding to the subsets $S_1, \ldots, S_m$ satisfying the conditions of eventown above. This means that

\begin{align}
      v_i \cdot v_i & = 0 \mod 2 \quad \text{for every } i \\
      v_i \cdot v_j & = 0 \mod 2 \quad \text{for every } i \neq j.
\end{align}

Now consider the subspace

$$U = \text{span}(v_1, \ldots, v_m)$$

and use (6)-(7) to show that (Exercise!)

$$U \subseteq U^\perp$$

which then implies $\dim(U) \leq \dim(U^\perp)$. This together with the Rank Theorem

$$n = \dim(V) = \dim(U) + \dim(U^\perp)$$

implies $\dim(U) \leq \lfloor n/2 \rfloor$, that is

$$U = \text{span}(u_1^*, \ldots, u_k^*) \quad \text{for } k \leq \lfloor n/2 \rfloor$$

This tells us that $|U| \leq 2^{\lfloor n/2 \rfloor}$ (Exercise!).
4 What to remember and where to look

The rank of a matrix tells us “how rich/complex” is that matrix. The proof for the jigsaw puzzle can be informally described in these terms:

If you want to build a “complex object” (high rank) using “simple objects” (low rank), then you need many of them.

When working with vectors it might be a good idea to “pack” them into a matrix and look at the rank to get a compact and elegant proof.

We have seen another way to prove upper bounds on “the number of objects” based on orthogonality and dimension of some subspace (an application to eventown).

The jigsaw puzzle with graphs (and its connection to the addressing problem – Section 2.3) is described in [BF92, Section 1.4].

References


A Linear Algebra: properties of the rank

In the following we fix an arbitrary field $\mathbb{F}$. We consider matrices $A, B, C \in \mathbb{F}^{n \times m}$ and linear independence of their columns (or rows) over $\mathbb{F}$.

**Definition 3.** The row rank (resp., column rank) of a matrix is the maximum number of rows (resp., columns) that are linearly independent.

Consider a generic matrix

$$A = \begin{pmatrix} | & | & | & | \\ c_1 & \cdots & c_m & | \\ | & | & | & | \\ \end{pmatrix} = \begin{pmatrix} - & r_1 & - \\ \vdots & - & - \\ - & r_n & - \\ \end{pmatrix}$$

Let $\alpha$ and $\beta$ be the row and the column rank of $A$, respectively. Then there are $\alpha$ rows (and $\beta$ columns) that “generate/span” all rows (all columns) of $A$:
For $\alpha$ and $\beta$ being the column and the row rank of $A$

\[
\text{span}(c_1, \ldots, c_m) = \text{span}(c_{j_1}, \ldots, c_{j_\alpha}) \tag{8}
\]

\[
\text{independent}
\]

\[
\text{span}(r_1, \ldots, r_n) = \text{span}(r_{i_1}, \ldots, r_{i_\beta}) \tag{9}
\]

\[
\text{independent}
\]

**Exercise 7.** Prove (9) and (8).

This theorem says that we can simply talk about the **rank of a matrix**:

**Theorem 4.** The row rank is equal to the column rank.

**Proof.** For any matrix

\[
A = \begin{pmatrix}
| & | & | \\
| c_1 & \cdots & c_m \\
| & | & |
\end{pmatrix} = \begin{pmatrix}
| & | & | \\
| -r_1 & - \\
| \vdots \\
| -r_n & - \\
| & | & |
\end{pmatrix}
\]

let $\alpha = \text{rowrank}(A)$ and thus $c_1, \ldots, c_m \in \text{span}(c_{i_1}, \ldots, c_{i_\alpha})$ by (8). Let us consider the matrix with only these $\alpha$ columns from $A$:

\[
A' = \begin{pmatrix}
| & | & | \\
| c_{i_1} & \cdots & c_{i_\alpha} \\
| & | & |
\end{pmatrix}
\]

Every $c_i$ can be written as a linear combination of these columns, that is $c_i = A' \lambda^i$ for some vector $\lambda^i$ of length $\alpha$. In the “language” of matrix multiplication we have

\[
A = \begin{pmatrix}
where $\lambda_i$ is the $i^{th}$ row of the matrix $\Lambda$. The Linear Algebra Bound (Theorem 1 in Lecture 2) and (9) imply $\beta \leq \alpha$. That is,

$$\text{rank}(A) \leq \text{colrank}(A)$$

for any matrix $A$. By applying this to the transpose $A^T$ we get the opposite inequality. \hfill $\square$

**Theorem 5.** $\text{rank}(A + B) \leq \text{rank}(A) + \text{rank}(B)$

**Proof.** We have our three matrices

$$\left( \begin{array}{c|c|c} c^1 & \cdots & c^m \\ \hline \end{array} \right) = \left( \begin{array}{c|c|c} a^1 & \cdots & a^m \\ \hline \end{array} \right) + \left( \begin{array}{c|c|c} b^1 & \cdots & b^m \\ \hline \end{array} \right)$$

Take a basis for the columns of $C$, of $A$ and of $B$

- $\hat{c}^1, \ldots, \hat{c}^{r_c}$ for $r_c = \text{rank}(C)$
- $\hat{a}^1, \ldots, \hat{a}^{r_a}$ for $r_a = \text{rank}(A)$
- $\hat{b}^1, \ldots, \hat{b}^{r_b}$ for $r_b = \text{rank}(B)$

Observe that

$$\hat{c}^1, \ldots, \hat{c}^{r_c} \in \text{span}(a^1, \ldots, a^m, b^1, \ldots, b^m) = \text{span}(\hat{a}^1, \ldots, \hat{a}^{r_a}, \hat{b}^1, \ldots, \hat{b}^{r_b})$$

Then Linear Algebra Bound (Theorem 1 in Lecture 2) implies $r_c \leq r_a + r_b$. \hfill $\square$
Exercise Set 3 – HS12  
(Linear Algebra Methods in Combinatorics)

You can submit solutions also by email by next lecture – 17.10.2011. These exercises are non-graded but you get feedback on your submitted solutions.

This is about the “application” of the jigsaw problem with graph. You may want to have a look at Section 2.3.

**Exercise 1.** A \{0, 1, *\}-addressing for the complete graph is a labelling of the vertices

\[ l : V \to \{0, 1, *\}^m \]

such that

\[ d_H(l(u), l(v)) = 1 \quad \text{for every } u \neq v \]

where \(d_H()\) is a **modified Hamming distance** where ‘*’ is a jolly joker (for instance \(d_H(*10, 110) = 0, d_H(1 * 0, 011) = 2, \text{ etc.}\) – two coordinates are different only if one is 0 and the other is 1, and the distance between two labels is the number of coordinates in which they differ. The length of this addressing is \(m\), the number of “bits” of the labels. Prove that \(m \geq n - 1\) by reducing this problem to the Jigsaw Puzzle with Graphs in the lecture.

For this exercise you may want to look back at the proof in Section 2.2.

**Exercise 2** (Jigsaw puzzle with graphs – a wrong proof). The matrices \(A_1, \ldots, A_m\) associated to our bipartite graphs are now obtained by orienting the edges in a “smart” way:

\[
\begin{array}{ccc}
1 & 4 \\
2 & 3
\end{array}
\]

that is always go from node \(i\) to node \(j > i\). Then the sum of all these matrices, \(S = A_1 + \cdots + A_m\), is like this

\[
\begin{pmatrix}
0 & \cdots & 1 \\
0 & \cdots & 0
\end{pmatrix}
\]
and it is immediate to see that \( \text{rank}(S) = n - 1 \). We conclude that

\[
n - 1 \leq \text{rank}(S) \leq \text{rank}(A_1) + \cdots + \text{rank}(A_m) = m
\]

Show that this proof is not correct (which of these steps are correct and what are wrong?).

Two exercises on orthogonality and dimension.

**Exercise 3.** Consider a subspace \( U = \text{span}(u_1, \ldots, u_k) \) of a vector space \( V = \mathbb{F}^n \). Prove that a vector \( w \in U^\perp \) if and only if it is orthogonal to these \( k \) elements,

\[
w \perp u_1, w \perp u_2, \ldots, w \perp u_k
\]

(*Hint:* let us adopt the notation \( \langle , \rangle \) for the inner product and use linearity of the inner product:

\[
\langle u + v, w \rangle = \langle u, w \rangle + \langle v, w \rangle \quad \text{and} \quad \langle \lambda v, w \rangle = \lambda \langle u, w \rangle
\]

So with this notation \( u \perp v \iff \langle u, v \rangle = 0 \).)

**Exercise 4.** Prove the following result:

**Theorem.** Let \( U \) be a subspace of a linear space \( V \). If \( v_1, \ldots, v_m \in U \) are linearly independent then \( m \leq \dim(U) \).

The next exercise does not use results from this lecture.

**Exercise 5.** I construct a graph \( G = (V, E) \) as follows: The vertices are (all) the subsets of \([n]\) which have cardinality 3, and an edge exists if and only if the two corresponding subsets \((S_i \text{ and } S_j)\) have intersection \((S_i \cap S_j)\) of even cardinality. Prove that this graph has no clique (complete subgraph) with \( n + 1 \) nodes.